

A dark blue vertical bar is on the left. A blue arrow points right from it, containing the date.

9/22/2022

Washing Machine

Finite State Machine

Several thin, curved, light grey lines sweep upwards from the bottom left towards the center of the page.

Annadata, Yagna Srinivasa Harsha

yxa210024

Duvvuri, Sai Pravallika

psd220003

Thota, Sai Abhishek

sxt210056

Verilog Code and Test bench

```
`timescale 1ns / 1ps

////////////////////////////////////

// Module Name: washingmachine

////////////////////////////////////

module washingmachine(clk, reset, closeddoor, startmachine, filledmachine,
detergentadded, machinecycle_timeout,
waterdrained, spin_timeout_check, doorlock_check, motoron_check,
fillvalue_check,          drainvalue_check,          done,
detergent_wash, checkwater_wash); input clk, reset; input [255:0]
closeddoor;   input   [127:0]   startmachine;   input   [127:0]
filledmachine; input   [127:0]   detergentadded; input   [127:0]
machinecycle_timeout; input [127:0] waterdrained; input [127:0]
spin_timeout_check; output reg [127:0] doorlock_check; output reg
[127:0] motoron_check; output reg [127:0] fillvalue_check; output
reg [127:0] drainvalue_check; output reg [127:0] done; output reg
[127:0] detergent_wash; output reg [127:0] checkwater_wash;

//defining the states parameter checkdoor
= 3'b000; parameter fillwater = 3'b001;
parameter detergentadder = 3'b010;
parameter cycling = 3'b011; parameter
waterdraining = 3'b100; parameter spinning
= 3'b101; reg[2:0] currentstatecheck,
nextstatecheck;

always@(currentstatecheck or startmachine or closeddoor or filledmachine
or detergentadded or waterdrained or machinecycle_timeout
or spin_timeout_check) begin case(currentstatecheck)
    //checking the door if it is locked and closed.
    //machine is started at this stage.
    checkdoor: if(startmachine==1'b1 &&
closeddoor==1'b1) begin nextstatecheck =
fillwater; motoron_check = 1'b0;
fillvalue_check = 1'b0; drainvalue_check
= 1'b0; doorlock_check = 1'b1;
detergent_wash = 1'b0; checkwater_wash =
1'b0; done = 1'b0; end else begin
```

```

nextstatecheck = currentstatecheck;
motoron_check = 1'b0; fillvalue_check =
1'b0; drainvalue_check = 1'b0;
doorlock_check = 1'b0; detergent_wash =
1'b0; checkwater_wash = 1'b0; done =
1'b0; end

//Water is filled for soaking the clothes.
fillwater: if (filledmachine==1'b1) begin
if(detergent_wash == 1'b0) begin
nextstatecheck = detergentadder;
motoron_check = 1'b0; fillvalue_check =
1'b0;
drainvalue_check = 1'b0;
doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b0; done =
1'b0; end else begin nextstatecheck
= cycling; motoron_check = 1'b0;
fillvalue_check = 1'b0;
drainvalue_check = 1'b0;
doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b1; done =
1'b0; end end else begin
nextstatecheck = currentstatecheck;
motoron_check = 1'b0;
fillvalue_check = 1'b1;
drainvalue_check = 1'b0;
doorlock_check = 1'b1;
detergent_wash = 1'b0;
checkwater_wash = 1'b0; done =
1'b0; end

//detergent/soap will be added. detergentadder:
if(detergentadded==1'b1)
begin nextstatecheck =
cycling; motoron_check =

```

```

1'b0; fillvalue_check =
1'b0; drainvalue_check =
1'b0; doorlock_check =
1'b1; detergent_wash =
1'b1; checkwater_wash =
1'b0; done = 1'b0; end
else begin nextstatecheck
= currentstatecheck;
motoron_check = 1'b0;
fillvalue_check = 1'b0;
drainvalue_check = 1'b0;
doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b0;
done = 1'b0; end

//Washing cycle for the cleaning the clothes
cycling: if(machinecycle_timeout == 1'b1)
begin nextstatecheck = waterdraining;
motoron_check = 1'b0; fillvalue_check = 1'b0;
drainvalue_check = 1'b0; doorlock_check =
1'b1; detergent_wash = 1'b1; checkwater_wash
= 1'b1; done = 1'b0; end else begin
nextstatecheck = currentstatecheck;
motoron_check = 1'b1; fillvalue_check = 1'b0;
drainvalue_check = 1'b0;
doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b1;
done = 1'b0; end

//draining the water from wet clothes waterdraining:
if(waterdrained==1'b1) begin
if(checkwater_wash==1'b0) begin
nextstatecheck = fillwater;
motoron_check = 1'b0;
fillvalue_check = 1'b0;
drainvalue_check = 1'b0;

```

```

doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b1; done =
1'b0; end else begin nextstatecheck
= spinning; motoron_check = 1'b0;
fillvalue_check = 1'b0;
drainvalue_check = 1'b0;
doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b1; done =
1'b0; end end else begin
nextstatecheck = currentstatecheck;
motoron_check = 1'b0;
fillvalue_check = 1'b0;
drainvalue_check = 1'b1;
doorlock_check = 1'b1;
detergent_wash = 1'b1;
checkwater_wash = 1'b1; done =
1'b0;
end

//drying the clothes which are drained.
spinning: if(spin_timeout_check==1'b1)
begin nextstatecheck = checkdoor;
motoron_check = 1'b0; fillvalue_check =
1'b0; drainvalue_check = 1'b0;
doorlock_check = 1'b1; detergent_wash =
1'b1; checkwater_wash = 1'b1; done =
1'b1; end else begin nextstatecheck =
currentstatecheck; motoron_check =
1'b0; fillvalue_check = 1'b0;
drainvalue_check = 1'b1; doorlock_check
= 1'b1; detergent_wash = 1'b1;
checkwater_wash = 1'b1; done = 1'b0;
end default: begin nextstatecheck =
checkdoor; motoron_check = 1'b0;

```

```

    fillvalue_check = 1'b0;
drainvalue_check = 1'b0;
doorlock_check = 1'b1; detergent_wash
= 1'b1; checkwater_wash = 1'b1; done
= 1'b1; end endcase end
always@(posedge clk or posedge reset)
begin if(reset) begin
currentstatecheck<=3'b000; end else
begin
currentstatecheck<=nextstatecheck;
end end endmodule

```

Test bench

```

`timescale 1ns / 1ps
/////////////////////////////////////////////////////////////////
// Module Name: washing_tb
/////////////////////////////////////////////////////////////////
//// module WB_TB(); reg clk, reset; reg [255:0] closeddoor; reg
[127:0] startmachine; reg [127:0] filledmachine; reg [127:0]
detergentadded; reg [127:0] machinecycle_timeout; reg [127:0]
waterdrained; reg [127:0] spin_timeout_check; wire [127:0]
doorlock_check;

wire [127:0] motoron_check;
wire [127:0] fillvalue_check;
wire [127:0]
drainvalue_check; wire
[127:0] done; wire [127:0]
detergent_wash; wire [127:0]
checkwater_wash;

washingmachine machine1(clk, reset, closeddoor, startmachine,
filledmachine, detergentadded,machinecycle_timeout,
waterdrained, spin_timeout_check, doorlock_check,
motoron_check, fillvalue_check, drainvalue_check,
done,detergent_wash, checkwater_wash); initial begin clk =
8'b0; reset = 8'b1; startmachine = 8'b0; closeddoor = 8'b0;
filledmachine = 8'b0; waterdrained = 8'b0; detergentadded =
8'b0; machinecycle_timeout = 8'b0; spin_timeout_check = 8'b0;

```

```
#10 reset=8'b0;
#10 startmachine=8'b1;closedoor=8'b1;
#20 filledmachine=8'b1;
#20 detergentadded=8'b1;
#20 machinecycle_timeout=8'b1;
#20 waterdrained=8'b1; #20
spin_timeout_check=8'b1; end
always begin #10 clk = ~clk;
end endmodule
```

Behavioral and Synthesized waveform

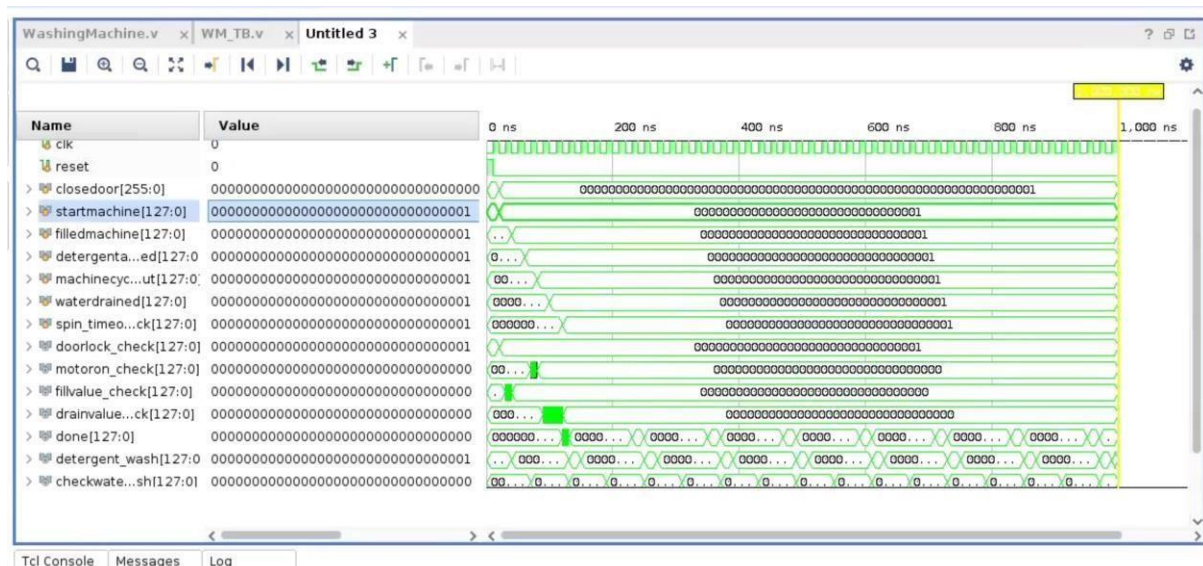
As Finite State Machines are used to create many real-life appliances, as a part of it we have implemented the design of the washing machine which used 6 finite states. The 6 finite states

are as follows

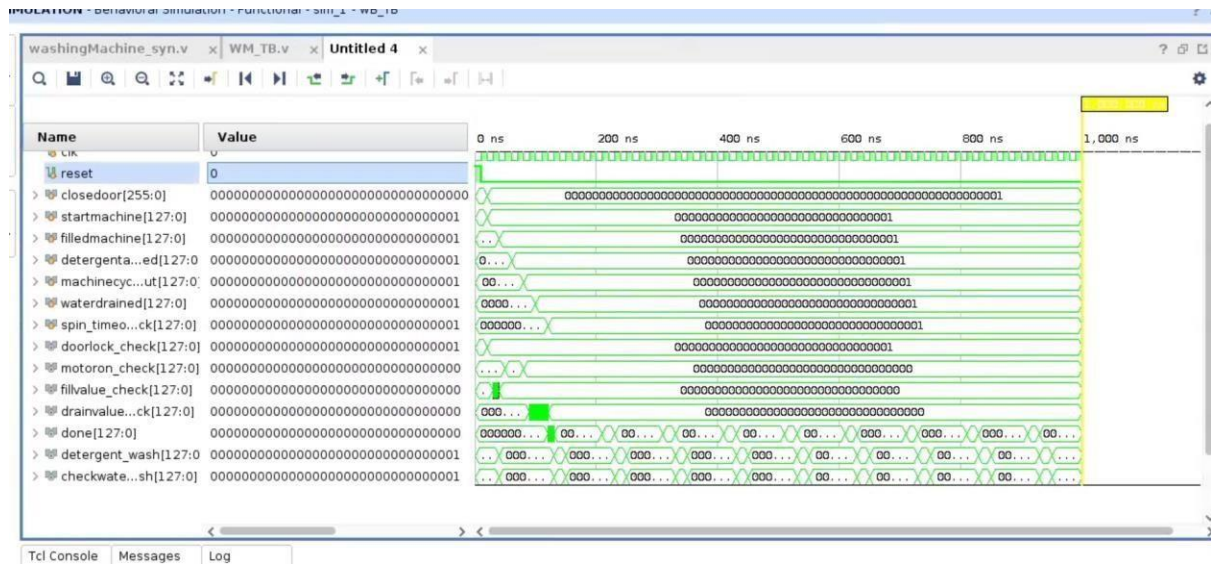
- Closing the door:
 - o Washing machine starts when the door is closed and locked.
- Filling the water:
 - o Washing machine is filled with water which is used for soaking the clothes.
- Adding detergent/soap:
 - o Washing machine uses detergent/soap to clean and wash the clothes.
- Cycle for wash:
 - o Cycle for wash is used by the washing machine to wash the clothes for certain time.
- Draining:
 - o Wet clothes will be drained after washing and cleaning them.
- Spinning:
 - o Spinning is used for drying the wet clothes.

There are 9 inputs and 7 outputs in this finite state design. It also contains about 6 stage parameters. To toggle between the states, we have used switch case scenarios. There are additional 2 registers namely current and next states which are used in the code.

Behavioral waveform

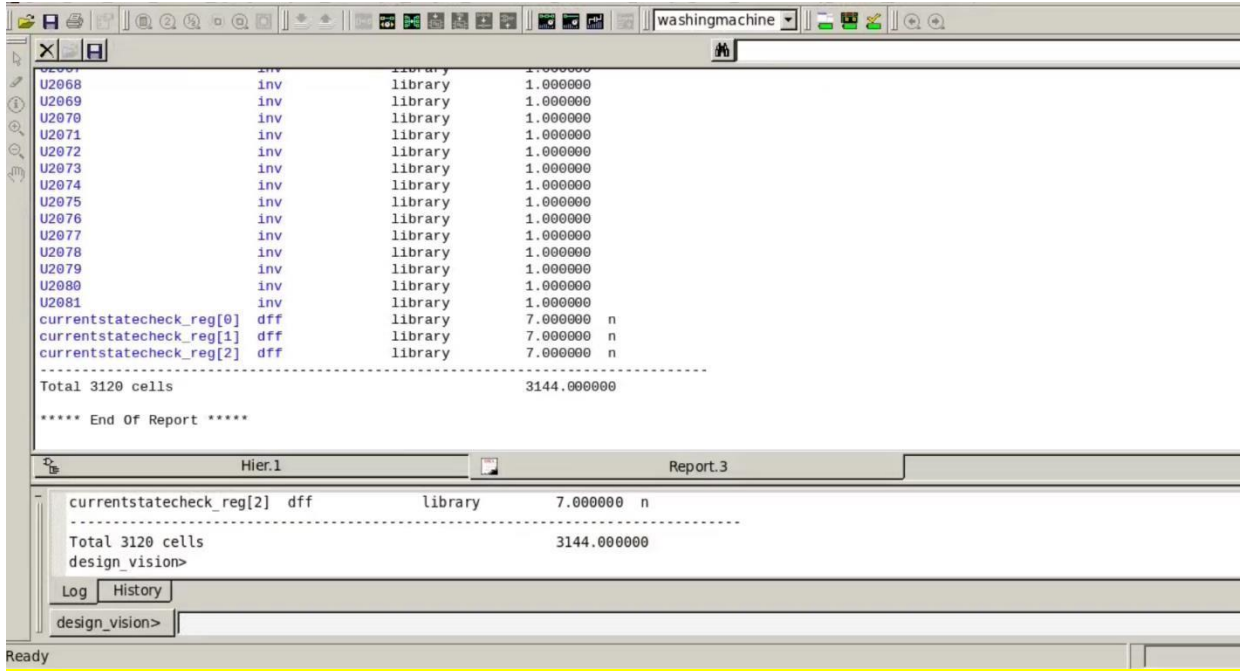


Synthesized waveform

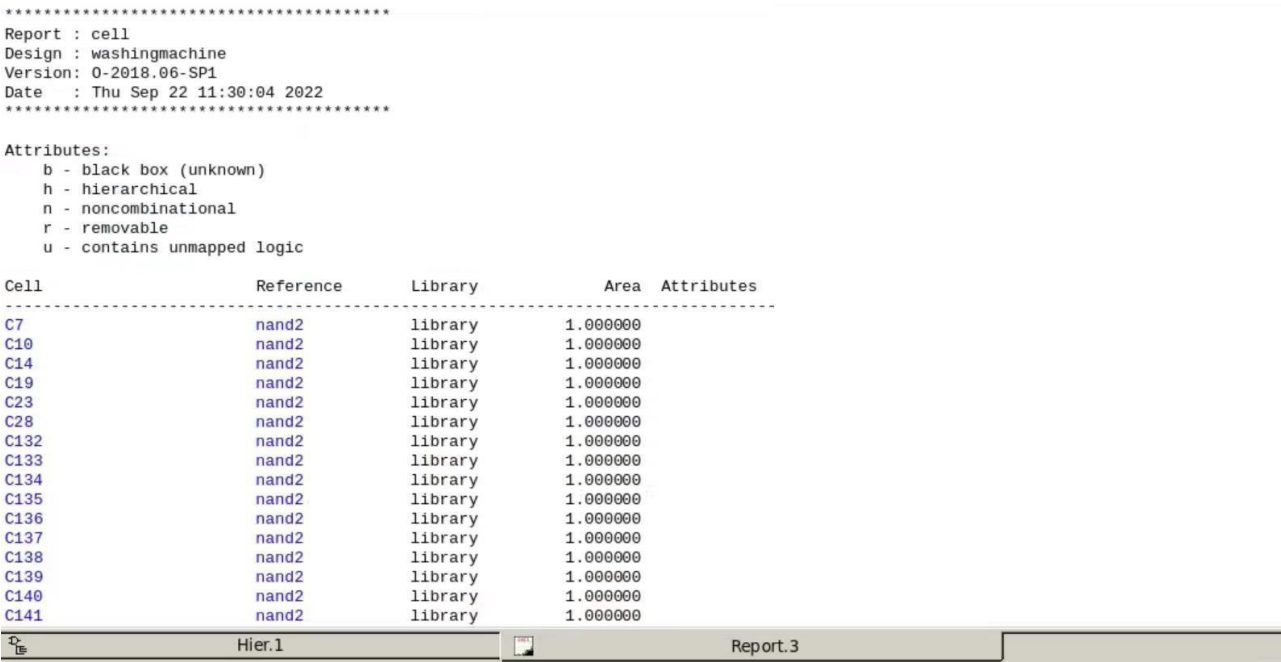


As observed, we were able to successfully generate the same waveforms. All the 6 stages are clearly generated in both the behavioral and synthesized waveforms.

Cell count



The above screenshot shows total of 3120 cells



The above screenshot shows the start process

U10	inv	library	1.000000
U11	nand3	library	1.000000
U12	nand2	library	1.000000
U13	nand2	library	1.000000
U14	nand3	library	1.000000
U15	nor2	library	1.000000
U16	nor3	library	1.000000
U17	inv	library	1.000000
U18	nand4	library	1.000000
U19	aoi22	library	2.000000
U20	nand2	library	1.000000
U21	nand4	library	1.000000
U22	aoi12	library	2.000000
U23	inv	library	1.000000
U24	inv	library	1.000000
U25	nor2	library	1.000000
U26	nor2	library	1.000000
U27	inv	library	1.000000
U28	aoi22	library	2.000000
U29	nand3	library	1.000000
U30	inv	library	1.000000
U31	nor3	library	1.000000
U32	nor2	library	1.000000
U33	inv	library	1.000000
U34	aoi22	library	2.000000
U35	nand3	library	1.000000
U36	inv	library	1.000000
U37	nand2	library	1.000000
U38	nand2	library	1.000000
U39	inv	library	1.000000
U40	nand4	library	1.000000
U41	inv	library	1.000000

Hier.1

Report.3

Use mouse cursor to tool used to highlight objects.

currentstatecheck_reg[0]	dff	library	7.000000	n
currentstatecheck_reg[1]	dff	library	7.000000	n
currentstatecheck_reg[2]	dff	library	7.000000	n

Total 3120 cells				3144.000000

The above screenshot shows the DFF units and other gates

Conclusion:

As observed, we were able to successfully generate the same waveforms. All the 6 stages are clearly generated in both the behavioral and synthesized waveforms.